7. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age=father’s age.

```
import java.util.Scanner;

class WrongAgeException extends Exception {
    public WrongAgeException(String message) {
        super(message);
    }
}

class Father {
    int age;
    public Father(int age) throws WrongAgeException {
        if (age < 0) {
            throw new WrongAgeException("Age cannot be negative");
        }
        this.age = age;
    }
}

class Son extends Father {
    int sonAge;
    public Son(int fatherAge, int sonAge) throws WrongAgeException {
        super(fatherAge);
        if (sonAge < 0) {
            throw new WrongAgeException("Age cannot be negative");
        }
        if (sonAge >= fatherAge) {
            throw new WrongAgeException("Son's age cannot be greater than father's age");
        }
        this.sonAge = sonAge;
    }
}

 class Demo {
    public static void main(String[] args) {
        try {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter father's age: ");
            int fage = sc.nextInt();
            Father father = new Father(fage);

            System.out.println("Enter son's age: ");
            int sage = sc.nextInt();
            Son son = new Son(father.age, sage);

            System.out.println("Father's age: " + father.age);
            System.out.println("Son's age: " + son.sonAge);
        } catch (WrongAgeException e) {
            System.out.println("Exception caught: " + e.getMessage());
        }
    }
}
```

```
C:\Users\sammj\OneDrive\Desktop\JAVA LAB\lab 7>java Demo
Enter father's age:
-40
Exception caught: Age cannot be negative

C:\Users\sammj\OneDrive\Desktop\JAVA LAB\lab 7>java Demo
Enter father's age:
40
Enter son's age:
-20
Exception caught: Age cannot be negative

C:\Users\sammj\OneDrive\Desktop\JAVA LAB\lab 7>java Demo
Enter father's age:
40
Enter son's age:
50
Exception caught: Son's age cannot be greater than father's age

C:\Users\sammj\OneDrive\Desktop\JAVA LAB\lab 7>
```

7. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor, which takes the age and throws the exception WrongAge() when the input age <0. In son class, implement a constructor that calls both father and son age and throws exception if son age > fathers age.

```
class WrongAgeException extends Exception
{
    public WrongAgeException(String message)
    {
        super(message);
    }
}
class Father
{
    int age;
    public father(int age) extends throws WrongAgeException
    {
        if (age <0)
        {
            throw new WrongAgeException("Age cannot be negative");
        }
        this.age = age;
}   }
class son extends Father
    {
        int sonAge;
        public Son(int fatherAge, int sonAge) throws
        WrongAgeException
        {
            if (sonage <0)
            {
                throw new WrongException("Age cannot be negative");
            }
            if (sonage >= fatherage)
            { throw newWrongException("Sons age
                cannot be greater than fathers age");
            }
            this.sonAge = sonAge;
    }
```

```
public class Demo {

    public static void main (String[] args)
    {
        try
        {
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter father's age :");
            int fage = sc.nextInt()
            Father father = new Father(fage);
            int
            System.out.println("Enter son's age :");
            Son son = new Son(fage, 8);
            sage = sc.nextInt()
            Son son = new Son(fage, sage);
        }
        catch (WrongAgeException e)
        {
            System.out.println("Exception caught!", e.getMessage);
        }
    }
}
```

Output:

Enter father's age: -10

Age Exception caught: WrongAgeException: Father Age cannot be -ve

Enter father's age: 50

Enter son's age: -10

Exception caught: WrongAgeException: Son Age cannot be -ve

23/12/24

Enter father's age: 50

Enter son's age: 60

Exception caught: WrongAgeException: Son's age cannot be greater than father's